REMARKS

Claims 1-8, 10-18, 20-27 and 29-35 are currently pending in the subject application and are presently under consideration.

Favorable reconsideration of the subject patent application is respectfully requested in view of the comments and amendments herein.

I. Rejection of Claims 1-8, 10-18, 20-27 and 29-35 Under 35 U.S.C. §103(a)

Claims 1-8, 10-18, 20-27 and 29-35 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Renaud *et al.* (U.S. 5,958,051), in view of Shaw (U.S. App. 2002/0026634), and further in view of Graunke *et al.* (U.S. 5,991,399) and Evans *et al.* (U.S. 5,805,899). It is respectfully requested that this rejection be withdrawn for at least the following reasons. The cited references,, alone or in combination, fail to teach or suggest all limitations recited the subject claims..

> To reject claims in an application under §103, an examiner must establish a *prima facie* case of obviousness. A *prima facie* case of obviousness is established by a showing of three basic criteria. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, ***to modify the reference or to combine reference teachings***. Second, there must be a reasonable expectation of success. Finally, ***the prior art reference (or references when combined) must teach or suggest all the claim limitations.*** *See* MPEP §706.02(j). The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art and not based on applicant's disclosure. *See In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991) (emphasis added).

The subject invention relates to validating that correct modules and assemblies are used for an application in a dynamically linked runtime environment. An assembly can consist of modules and other assemblies. A manifest can contain a list of modules and/or assemblies. A hash of modules and manifests are used to verify correct versions and identify changes to modules and assemblies. In particular, independent claim 1 (and similarly independent claims

10, 18, 22, 23, 27 and 30) recites ***providing an assembly manifest with a hash of a manifest of at least one other assembly that the assembly depends on.***

The Examiner concedes that Renaud *et al.* does not "disclose a manifest with a hash of a manifest of one referenced assembly of the list of referenced assemblies." (*See* Office Action dated April 21, 2005, pg. 8). However, he incorrectly contends that Renaud *et al.* teaches representing a manifest with a hashed representation (signature 322 – Fig. 3B). (*See* Office Action dated April 21, 2005, pg. 8). To the contrary, the hashed representation (*e.g.*, signature) is not a hash of a manifest, but rather a hash of a signature. Furthermore, Renaud *et al.* fails to teach an assembly manifest that contains a reference to a manifest of another assembly. The cited art teaches a manifest containing a list of modules. Thus, Renaud *et al.* does not teach or suggest that the manifest is provided with a hash of a manifest of the at least one referenced assembly as recited in the subject claims.

Shaw fails to make up for the aforementioned deficiencies of Renaud *et al.* Shaw relates to a system for secure downloading of data. (*See* Abstract). Shaw discloses a table of hashes of segments of application code. (*See* Para 0034). Accordingly, Shaw discloses a hash of a code segment or module, and does not teach a hash of a manifest of an assembly. As is known in the art, an assembly refers to a grouping of files (modules) necessary to perform a particular application, and modules are portion(s) of a computer program that are created to carry out a particular function within the application, and can be utilized alone or combined with other modules in connection with enabling proper operation of the particular application. Shaw discloses hashing the files or modules rather than the list of files or manifest. As discussed in the specification "hashing the manifest of the referenced assembly is sufficient because that manifest in turn includes hashes of all its constituent files." (*See* pg. 3 ll. 18-20). Hashing the entire assembly would require more processing and decrease efficiency.

Evans *et al.* fails to make up for the deficiencies of Renaud *et al.* and Shaw. Evans *et al.* relates to providing versioning information for a plurality of software objects. (*See* abstract). More particularly, Evans *et al.* utilizes a hash value that is generated from the name of a version using a conventional ELF hashing function. (*See* Col. 11, ll. 46-48). The Office Action dated October 10, 2005 asserts that Evans *et al* teaches an assembly manifest referencing another assemble manifest. However, the cited art merely teaches a manifest containing references to different modules and different versions of each module to account for version control. Evans *et*

al does not teach or suggest an assembly manifest referencing another assemble manifest let alone hashing a manifest of a referenced assembly.

Moreover, Graunke *et al.* fails to make up for the deficiencies of Renaud *et al.*, Shaw and Evans *et al.* Graunke *et al.* relates to a method for distributing private keys to user application programs to decrypt encrypted digital content. (*See* Abstract). The Examiner asserts that Graunke *et al.* teaches that "a manifest is a signed information structure submitted by a manufacturer for identifying references to other hashed assemblies being delivered for integrity checking." (*See* Office Action dated April 21, 2005, pg. 8). However, as described by Graunke *et al.* the manifest references digital objects (*e.g.*, modules), not assemblies. (*See* Fig. 3 and Col. 6 ll. 53-56). Furthermore, Graunke *et al.* teaches hashing the entire digital objects, rather than the hash of a manifest or list of files of an assembly.

The combination of the cited art fails to teach or suggest *providing an assembly manifest with a hash of a manifest of at least one other assembly that the assembly depends* on as in the claimed invention. In view of at least the foregoing comments, it is readily apparent that Renaud *et al.*, Shaw, Evans *et al.* and Graunke *et al.*, alone or in combination, do not make obvious applicants' invention as recited in independent claims 1, 10, 18, 22, 23, 27 and 30 (and dependent claims 2-8, 11-17, 20, 21, 24-26 and 29 which respectively depend there *et al.* from). Accordingly, this rejection should be withdrawn.

CONCLUSION

The present application is believed to be in condition for allowance in view of the above comments and amendments. A prompt action to such end is earnestly solicited.

In the event any fees are due in connection with this document, the Commissioner is authorized to charge those fees to Deposit Account No. 50-1063 [MSFTP119US].

Should the Examiner believe a telephone interview would be helpful to expedite favorable prosecution, the Examiner is invited to contact applicants' undersigned representative at the telephone number below.

Respectfully submitted,

AMIN & TUROCY, LLP

Himanshu S. Amin
Reg. No. 40,894

AMIN & TUROCY, LLP
24TH Floor, National City Center
1900 E. 9TH Street
Cleveland, Ohio 44114
Telephone (216) 696-8730
Facsimile (216) 696-8731